

IN THE CLAIMS

1. (Original) A method comprising:

providing prerequisite information regarding page sub-components of a graphical user interface (GUI) that are prerequisites to other page sub-components of the GUI; in response to a request for a destination page of the GUI and with reference to the prerequisite information, identifying one or more page sub-component prerequisites of page sub-components associated with the destination page;

determining whether one or more requirements of an instance of an identified page sub-component prerequisite have been satisfied by invoking a method of the instance that causes stored information regarding the state of the page sub-component prerequisite to be retrieved from a current environment/context; and

causing the destination page to be displayed with (1) content associated with the identified page sub-component prerequisite if any of its one or more requirements have not been satisfied and (2) content associated with those of the page sub-components having no page sub-component prerequisites or having page sub-component prerequisites whose requirements have all been satisfied.

2. ✓ (Original) The method of claim 1 wherein the prerequisite information is stored in a Java properties file.

3. (Original) The method of claim 1, further comprising supporting definition of hierarchical relationships of page sub-component prerequisites by:

iterating through each of the identified page sub-component prerequisites for a particular page sub-component associated with the destination page in a predetermined order until encountering the first page sub-component prerequisite that has one or more unsatisfied requirements; and

displaying the first page sub-component prerequisite of the identified page sub-component prerequisites before displaying a second page sub-component prerequisite of the identified page sub-component prerequisites that has one or more unsatisfied requirements and that is dependent upon the first page sub-component prerequisite according to the predetermined order.

4. (Original) The method of claim 1, wherein each page sub-component has a prerequisite property and the prerequisite information includes, for each page sub-component that has one or more page sub-component prerequisites, a string identifying the one or more page sub-component prerequisites.

5. (Currently Amended) The method of claim 1, wherein the prerequisite information is structured as a list of attribute-value pairs, and wherein ~~the~~a syntax for identifying a first sub-component, sub_1 , and a second sub-component, sub_2 , as prerequisites of a third sub-component, sub_3 , is substantially as follows:

$sub_3.prereq = sub_1\ sub_2.$

6. (Original) The method of claim 1, wherein the request for the destination page comprises a HyperText Transfer Protocol (HTTP) request, and wherein the page sub-components of the GUI are associated with web pages.

A 1

7. (Original) The method of claim 1, further comprising modifying the prerequisite information without necessitating recompilation of software code.

8. (Original) The method of claim 1, wherein said determining whether one or more requirements of an instance of an identified page sub-component prerequisite have been satisfied includes requesting that a page sub-component prerequisite object verify whether all its requirements have been satisfied.

9. (Original) The method of claim 1, wherein page sub-component objects corresponding to the page sub-components of the GUI and page sub-component prerequisite objects responsible for ensuring satisfaction of one or more prerequisite conditions are loosely coupled and may be dynamically associated with each other by way of the prerequisite information.

10. (Original) A method of presenting a page requested by a user comprising:
in response to a request for a destination page of a graphical user interface (GUI), creating an instance of a container to represent the destination page, the container including a list of sub-components to render;
identifying one or more sub-components associated with the destination page; for each of the one or more sub-components
determining whether the sub-component has any page sub-component prerequisites with reference to a set of prerequisite information, the set of prerequisite information including information regarding sub-components of the GUI that are prerequisites to other sub-components of the GUI, and

if the sub-component has a page sub-component prerequisite and if one or more requirements of the page sub-component prerequisite remains unsatisfied, then adding an instance of the page sub-component prerequisite to the list of sub-components associated with the container, otherwise adding an instance of the sub-component to the list of sub-components; and

causing the destination page to be displayed by rendering the output of the instances on the list of sub-components, whereby page sub-component prerequisites that have one or more requirements that remain unsatisfied are displayed in place of the corresponding sub-components.

11. (Original) A graphical user interface (GUI) system for enforcing page sub-component prerequisites comprising:

a properties data store including information regarding page sub-components of the GUI that are prerequisites to other page sub-components of the GUI;

a base agent to respond to requests for a destination page of the GUI, in response to a request for the destination page, the base agent creating an instance of a container to represent the destination page and initiating display of the destination page after a list of page sub-components of the container has been populated; and

a sub-component prerequisite factory decoupling the page sub-components from their respective page sub-component prerequisites, the sub-component prerequisite factory to either (1) cause an instance of an identified page sub-component prerequisite to be added to the list of page sub-components if it determines that one or more requirements of the identified page sub-component prerequisite are unsatisfied or (2) cause an instance of the page sub-component to be added to the list of page sub-components, whereby page sub-component prerequisites that have

one or more unsatisfied requirements are displayed in place of the corresponding page sub-components.

12. (Currently Amended) The system of claim 11, wherein the ~~prerequisite information is stored in properties data store is~~ a Java properties file.

13. (Original) The system of claim 11, wherein the sub-component prerequisite factory supports hierarchical relationships of page sub-component prerequisites by:

A | iterating through each of the identified page sub-component prerequisites for a particular page sub-component associated with the destination page in a predetermined order until encountering the first page sub-component prerequisite that has one or more unsatisfied requirements; and

displaying the first page sub-component prerequisite of the identified page sub-component prerequisites before displaying a second page sub-component prerequisite of the identified page sub-component prerequisites that has one or more unsatisfied requirements and that is dependent upon the first page sub-component prerequisite according to the predetermined order.

14. (Original) The system of claim 11, wherein each page sub-component has a prerequisite property and the prerequisite information includes, for each page sub-component that has one or more prerequisite sub-components, a string identifying the one or more page sub-component prerequisites.

15. (Currently Amended) The system of claim 11, wherein at least a portion of the information of the properties data store is structured as a list of attribute-value pairs, and wherein ~~the~~ syntax for identifying a first sub-component, sub₁, and a second sub-component, sub₂, as prerequisites of a third sub-component, sub₃, is substantially as follows:

sub₃.prereq = sub₁ sub₂.

16. (Original) The system of claim 11, wherein the requests correspond to HyperText Transfer Protocol (HTTP) requests, and wherein the page sub-components of the GUI are associated with web pages.

17. (Original) The system of claim 11, wherein prerequisite relationships among two or more page sub-components of the page sub-components of the GUI may be modified without necessitating recompilation of software code by editing the information of the properties data store.

18. (Original) The system of claim 11, further comprising page sub-component objects corresponding to the page sub-components of the GUI and page sub-component prerequisite objects responsible for ensuring satisfaction of one or more prerequisite conditions are loosely coupled and may be dynamically associated with each other by way of the prerequisite information.

19. (Original) A machine-readable medium having stored thereon data representing sequences of instructions, the sequences of instruction which, when executed by a processor, cause the processor to:

identify one or more page sub-component prerequisites of page sub-components associated with a destination page of a graphical user interface (GUI) in response to a request for the destination page and with reference to the prerequisite information regarding page sub-components of the GUI that are prerequisites to other page sub-components of the GUI;

A | determine whether one or more requirements of an instance of an identified page sub-component prerequisite have been satisfied by invoking a method of the instance that causes stored information regarding the state of the page sub-component prerequisite to be retrieved from a current environment/context; and

cause the destination page to be displayed with (1) content associated with the identified page sub-component prerequisite if any of its one or more requirements have not been satisfied and (2) content associated with those of the page sub-components having no page sub-component prerequisites or having page sub-component prerequisites whose requirements have all been satisfied.